

Memoria de la prueba técnica Minderest

Introducción

En esta memoria voy a intentar condensar todo el proceso de análisis, diseño e implementación de la prueba técnica.

El objetivo de la prueba era gestionar los datos de productos de distintos clientes y poder establecer relaciones de equivalencias entre los productos, basicamente decir que estos productos son el mismo.

He implementado unicamente lo que se pedía en el enunciado de la tarea. Es decir tiene una funcionalidad bastante limitada.

Tecnologías usadas

El código esta escrito solo con Java, he usado MariaDB (11.4.9) como base de datos y VSCode como IDE.

Modelo de datos

Cliente (id_cliente (BIGINT UNSIGNED, PK, AUTO_INCREMENT), nombre (VARCHAR(255), NOT NULL), codigo(VARCHAR(24), NOT NULL UNIQUE))

Producto (id_producto (BIGINT UNSIGNED, PK, AUTO_INCREMENT) cliente_id (BIGINT UNSIGNED, NOT NULL) → FK a cliente.id_cliente nombre (VARCHAR(255), NOT NULL))

Producto_equivalencia (id_equivalencia (BIGINT UNSIGNED, PK, AUTO_INCREMENT), producto_a (BIGINT UNSIGNED, NOT NULL) → FK a producto.id_producto, producto_b (BIGINT UNSIGNED, NOT NULL) → FK a producto.id_producto)

Relaciones

cliente (1) — producto (N)

producto (N) — producto (N) vía producto_equivalencia

Comentarios

He implentado la base de datos de forma literal a como la he entendido en el ejercicio.

Hacer la tabla producto_equivalencia de esta forma fue mi primera idea aunque la deseche a favor de una tabla para agrupar los productos dado que pensaba que las consultas serian mas simples, volví a esta tipo de agrupación mas directa de las equivalencias porque está mas alineada con lo que marca el ejercicio. Mi problema con esta forma de hacerlo era que no estaba seguro de como sacar los productos que no estaban directamente relacionados es decir ($A=B$, $B=C$) pero no se me ocurría como podía relacionar que $A=C$ sin hacerlo explícitamente por eso pensé en agruparlos por “categorías”, lo resolví al realizar una consulta en cascada para poder sacar los productos que están

relacionados al producto que estas buscando aunque no estén directamente señalando le, para esta consulta me ayude de copilot dado que no estoy tan familiarizado con esta clase de sintaxis, se que es posible y mas o menos tengo idea de como se enfoca la consulta pero si me ha hecho falta ayuda externa para poder hacerla.

Implementación en Java

Modelos de Entidad

Cliente y Producto, no hay mucho que decir clases que hacen espejo a su contraparte en la base de datos

Conexión

Tengo que decir que la forma de hacer la conexión ha sido un poco como un juego para mi, he imitado una implantación que tenia en otro lenguaje (php) lo máximo posible, no estoy seguro de que sea la mejor forma ni la mas elegante de realizar la conexión.

Sigue el patrón singleton y devuelve una única instancia de la conexión.

Carga db.properties donde se configuran los datos de conexión a la base de datos.

Servicio

Aquí se encuentra la lógica principal de la aplicación.

Todas las sentencias e inserciones se gestionan con sentencias preparadas donde se insertan los datos y en un trycatch se ejecutan, esto saca los errores para que sea mas facil el debug y de un poco de protección al solo permitir el tipo de dato que se haya puesto.

insertarProducto(Producto producto): Inserta el producto (instancia de la clase) a la base de datos en su correspondiente tabla.

crearEquivalencia(Producto productoA, Producto productoB) : Crea la equivalencia entre dos productos en la tabla de equivalencias, le he puesto validaciones bastante basicas, actualmente comprueba que no se ponga el mismo producto dos veces y que el cliente no es el mismo en ambos productos.

obtenerProductosEquivalentes(Long clienteID, Producto producto) : Usa la sentencia preparada para buscar en cascada los productos equivalentes de forma recursiva y devuelve un array de Productos. Como decía anteriormente aquí he usado ayuda de copilot para realizar la sentencia.

Adicionalmente he creado el metodo de listar para hacer mas rapido la comprobación de los cambios en las tablas y ver que todo funciona correctamente

Pruebas realizadas

Carga de esquema y datos: He ejecutado multiples veces Schemapruebatecnica.sql en MariaDB para verificar la creación de tablas y población de datos.

Consulta de equivalentes: En App.java con un producto de prueba he comprobado la funcionalidad de el metodo de obtenerProductosEquivalentes, insertarProductos y crearEquivalencias.

Actualmente esta comentado en el archivo de App.java pero la salida esperada es la siguiente: Con Producto(1L, "Leche entera marca A", 1L) lista de productos conectados a 1 (p. ej., 2, 3, 4, 14) excluyendo el propio y, si se pasa `clienteID = 1L`, filtrando los del cliente 1.

Se comprueba la inserción de equivalencia valida y el manejo de errores.